

# Semantic Web Technologies 1

Sebastian Rudolph und Elena Simperl

Wintersemester 2011/12

<http://semantic-web-grundlagen.de>

## Lösung der Übung 1: RDF und RDF Schema

---

### Lösung der Aufgabe 1.1

- (a) Ein Auszug aus einer früheren Startseite des AIFB ([www.aifb.uni-karlsruhe.de](http://www.aifb.uni-karlsruhe.de)):

```
<table width="100%">
<tr>
<td width="33%" align="left">
<a href="mailto:webmaster@aifb.uni-karlsruhe.de">
<font class="footnote">&copy; AIFB 2001-2005</font></a></td>
<td width="33%" align="center">
&nbsp;<td width="33%" align="right">
<a href="http://www.aifb.uni-karlsruhe.de/about.html">
<font class="footnote">About our SEmantic portAL</font></a></td>
</tr>
</table>
```

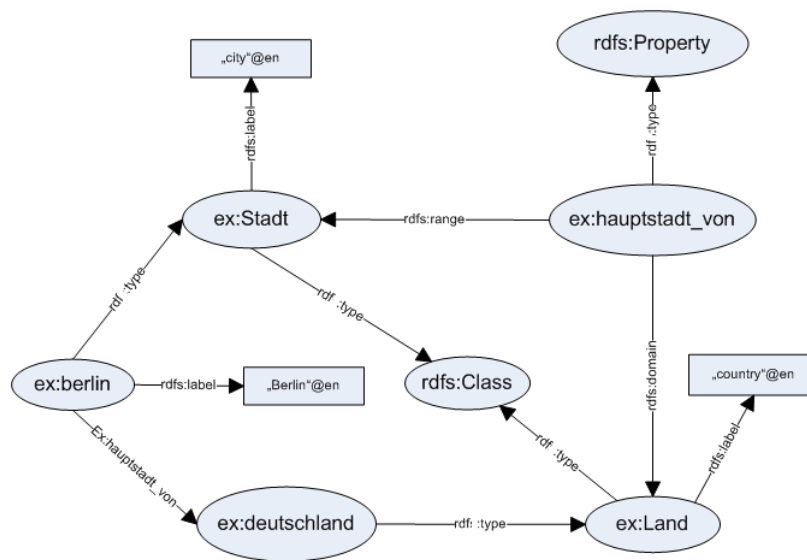
- (i) Linie 5,7: **&copy;** **&nbsp;**; DTD declaration fehlt.
- (ii) Linie 10: **</td>** fehlt.
- (b) Diese Dokument hält sich nicht an die Empfehlung, alle Tags klein zu schreiben:

```
<Liste laenge=4>
<Listeneintrag pos=1>
<Element/>Lorem ipsum<element/>
</Listeneintrag>
<listeneintrag pos=3/>
</Liste>
```

- (i) Linie 4,2, 5: “**<**” zeichen fehlen.
- (ii) Linie 4: **</Listeneintrag>** ist nicht das Gleich wie **<Listeneintrag>**

**Lösung der Aufgabe 1.2** see RDF(S) specification.

**Lösung der Aufgabe 1.3:(b)**



**Lösung der Aufgabe 1.3:(c)**

@prefix rdf: <http://www.w3.org/1990/02/22-rdf-syntax-ns#>.

@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#>.

@prefix ex: <http://example.org/>

ex:deutschland                      rdf:type                      ex:Land.

ex:hauptstadt\_von                  rdfs:type                      rdfs:Property;  
    rdfs:domain                      ex:Stadt;  
    rdfs:range                      ex:Land.

ex:Land                              rdf:type                      rdfs:Class;  
    rdfs:label                      "country"@en.

ex:berlin                            rdf:type                      ex:Stadt;  
    rdfs:label                      "Berlin"@en;  
    ex:hauptstadt\_von              ex:deutschland.

ex:Stadt                              rdf:type                      rdfs:Class;  
    rdfs:label                      "City"@en.

**Lösung der Aufgabe 1.4:**

```

<rdf:RDF
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
  xmlns:ex="http://example.org/">

  <rdf:Description rdf:about="http://example.org/vegetableThaiCurry">
    <ex:thaiGerichtBasierendAuf rdf:resource="http://example.org/kokomilsch"/>
  </rdf:Description>

  <rdf:Description rdf:about="http://example.org/sebastian">
    <rdf:type rdf:resource="http://example.org/Nussallergiker"/>
    <ex:isst rdf:resource="http://example.org/vegetableThaiCurry"/>
  </rdf:Description>

  <rdf:Description rdf:about="http://example.org/Nussallergiker">
    <rdfs:subClassOf rdf:resource="http://example.org/Bedauernswert"/>
  </rdf:Description>

  <rdf:Description rdf:about="http://example.org/thaiGerichtBasierendAuf">
    <rdfs:subPropertyOf rdf:resource="http://example.org/hatZutat"/>
    <rdfs:domain rdf:resource="http://example.org/Thaileandisch"/>
    <rdfs:range rdf:resource="http://example.org/Nussig"/>
  </rdf:Description>

  <rdf:Description rdf:about="http://example.org/hatZutat">
    <rdf:type rdf:resource="http://www.w3.org/2000/01/rdf-schema#
                                     ContainerMembershipProperty"/>
  </rdf:Description>

</rdf:RDF>

```

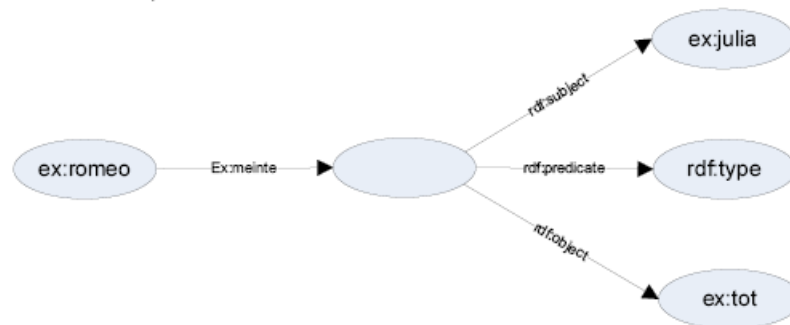
### Lösung der Aufgabe 1.5:

- (a) Leere Knoten können für beliebige Ressourcen stehen.  
♠ **falsch**: nicht für Prädikate – Prädikate müssen immer durch URI angegeben werden.
- (b) URLs können für beliebige Ressourcen stehen.  
♣ **wahr**
- (c) Jeder leere Knoten hat eine ID.  
♠ **falsch**: z.B. Der Wert Resource des Attributs rdf:parseType generiert automatisch einen neueren leeren Knoten ohne ID.

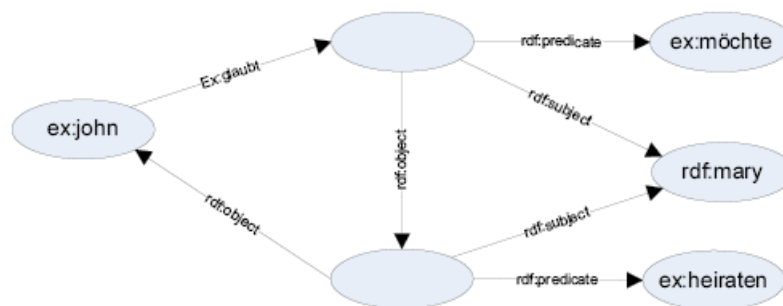
- (d) Zwei leere Knoten mit unterschiedlicher ID können für die gleiche Ressourcen stehen.  
♣ **wahr**
- (e) Zwei unterschiedlich URLs können für die gleiche Ressourcen stehen.  
♣ **wahr**
- (f) Kommen in mehreren RDF-Dokumenten leere Knoten mit der selben ID vor, dann müssen sie für die selbe Ressroucen stehen.  
♠ **falsch**: Eine ID von eine leere Knote ist lokal.
- (g) Kommen in mehreren RDF-Dokumenten die gleichen URIs vor, dann müssen sie für selbe Ressourcen stehen.  
♣ **wahr**: URI heisst "Unique Resource Identifier".
- (h) Zwei unterschiedliche Literale können niemals für den gleichen Wert stehen.  
♠ **falsch**: 2.0 und 2.00 stehen für gleichen wert in xsd:decimal.
- (i) Zwei Literal unterschiedlichen Datentyps können niemals für den gleichen Wert stehen.  
♠ **falsch**: 2 (xsd:integer) und 2.0 (xsd:decimal) stehen für den gleichen Wert d.h. 2.
- (j) Eine URI kann niemals für den Wert eines Datentyps stehen.  
♠ **falsch**: steht für den Wert des Datentyps xsd:anyURI.
- (k) Leere Knoten können nicht als Prädikate in Tripeln auftreten.  
♣ **wahr**
- (l) Leere Knoten können nicht für Propertys stehen (also für Ressourcen, welche der Klasse rdf:Property angehören).  
♠ **falsch**

### **Lösung der Aufgabe 1.7:**

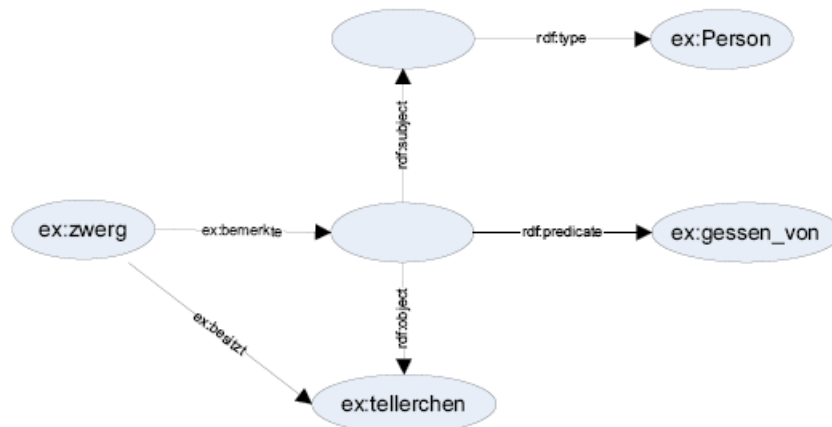
(a): Romeo meinte, Julia sei tot.



(b): John glaubt, dass Mary ihn heiraten möchte.



(c): Der Zwerg bemerkte, dass irgend jemand con seinem Tellerchen gegessen hatte.



**Lösung der Aufgabe 1.8:** (b),(c), und (e): modellieren nicht möglich in RDF(S).

@prefix rdf: <http://www.w3.org/1990/02/22-rdf-syntax-ns#>.  
 @prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#>.  
 @prefix ex: <http://example.org/>

- Jede Pizza ist eine Speise.

`ex:Pizza        rdfs:subClassOf        ex:Speise.`

- Alles, was einen Belag hat, ist eine Pizza.

`ex:hatBelag        rdfs:Domain        ex:Tomaten.`

- "Einen Belag haben" ist eine Enthaltenseins-Beziehung.

`ex:belagHaben        rdf:type        rdfs:ContainerMembershipProperty.`